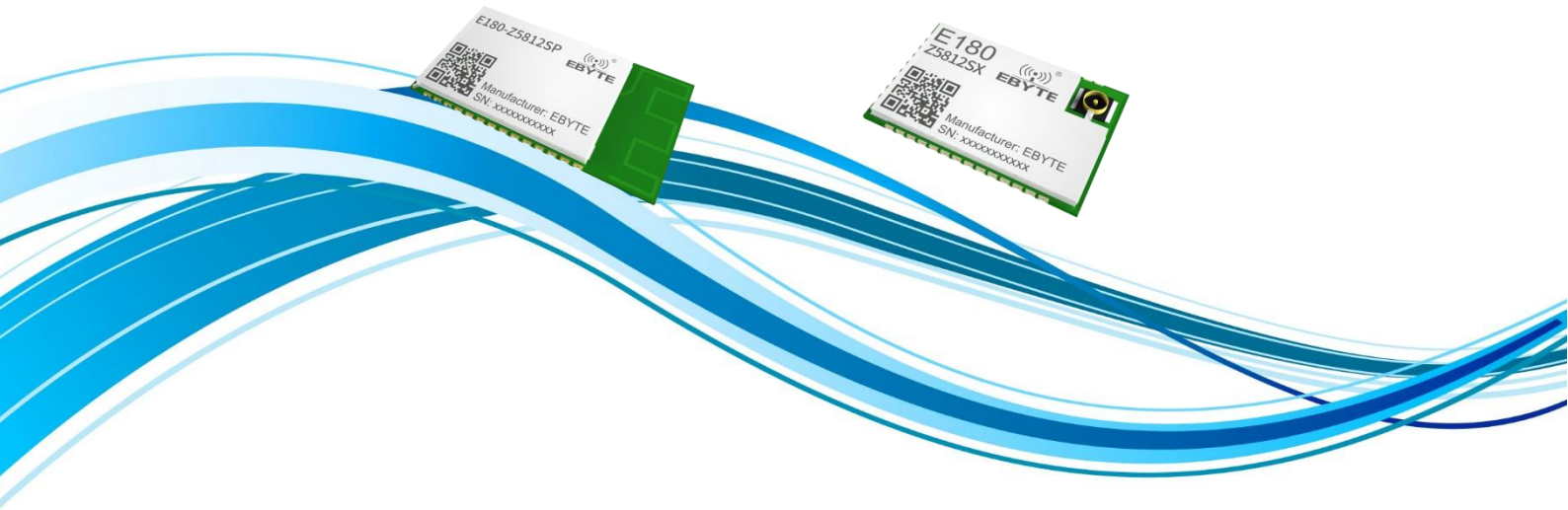
# E180-Z5812 Series User Manual

## TLSR8258 2.4GHz ZigBee3.0 multi-function SoC wireless module




Chengdu Ebyte Electronic Technology Co.,Ltd.

## Contents


Disclaimer ..... 6
1 Overview ..... 1
1.1 Product introduction ..... 1
1.2 Functional features ..... 1
1.3 Introduction to equipment types ..... 2
1.3.1 Coordinator ..... 2
1.3.2 Router ..... 3
1.3.3 Non-sleeping terminal ..... 3
1.3.4 Sleep terminal ..... 3
1.4 Application scenarios ..... 3
2 Specifications ..... 4
2.1 Limit parameters ..... 4
2.2 Working parameters ..... 4
2.3 Performance parameters ..... 5
3. Mechanical Dimensions and Pin Definitions ..... 6
4 Serial Port Input and Output ..... 10
4.1 HEX instruction mode ..... 10
4.1.1 HEX command frame format ..... 10
4.1.2 HEX instruction classification ..... 10
4.1.3 HEX commands supported by this product ..... 12
4.2 Universal mode ..... 14
4.2.1 Precautions for universal mode ..... 14
4.2.2 Universal mode schematic block diagram ..... 15
4.2.3 Universal mode usage tutorial ..... 16
4.3 Transparent transmission mode ..... 16
4.3.1 ZCL specification for data transparent transmission ..... 16
4.3.2 Target setting of data transparent transmission ..... 16
4.3.3 Binding transparent transmission target ..... 17
4.4 AT command mode ..... 17
4.4.1 Execution type ..... 17
4.4.2 Query formula ..... 18
4.4.3 Setting type ..... 18
4.4.4 AT command directory ..... 18
4. 4.5 New AT command ..... 18
4.5 Mode switching ..... 19
4.5.1 Mode switching table ..... 19
4.5.2 Precautions for mode switching ..... 20
5 Module Network Distribution and Data Transmission ..... 21
5.1 Module network distribution ..... 21
5.1.1 HEX command network configuration ..... 21
5.1.2 AT command network distribution ..... 21
5.1.3 Button configuration ..... 21
5.2 Data transmission ..... 21

5.2.1 Broadcast mode ..... 22
5.2.2 On-demand mode ..... 22
5.2.3 Multicast mode ..... 22
5.2.3 Binding transfer mode ..... 22
5.3 Set up binding ..... 22
5.3.1 Coordinator remote configuration ..... 22
5.3.2 Set binding with one click ..... 23
5.3.3 View bindings ..... 23
6 Universal Mode Usage Tutorial ..... 25
6.1 Routine 1: Simulate a three-way switch to access the gateway and use APP to control the switch ..... 25
6.1.1 Create the first switch port and add properties ..... 25
6.1.2 Create other switch ports and add properties ..... 28
6.1.3 Initialize attribute values ..... 29
6.1.4 Access gateway ..... 29
6.1.5 Use APP to control simulated 3-way switch ..... 30
6.2 Example 2: Simulating human body infrared sensor access gateway ..... 32
6.2.1 Create sensor port and add related properties ..... 32
6.2.2 Set attribute initialization values and access the gateway ..... 33
6.2.3 Sensor triggers alarm ..... 34
7 FAQ ..... 36
7.1 Transmission distance is not ideal ..... 36
7.2 Modules are easily damaged ..... 37
7.3 Bit error rate is too high ..... 37
Revise history ..... 37
About us ..... 38

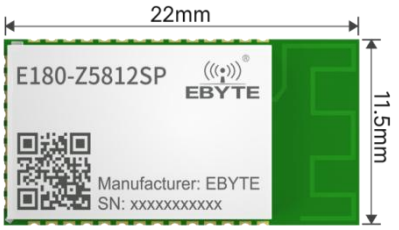
Disclaimer

# 1 Overview

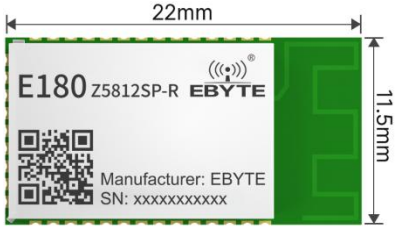
## 1.1 Product introduction

The E180-Z5812 series is a ZIGBEE module designed and produced by Chengdu Ebyte based on TELINK TLSR8258 wireless SOC with small size, low power consumption, high reliability, working in the 2.4GHz frequency band. The chip comes with a 32-bit high-performance MCU up to 48Mhz , the transmit power can reach up to 12dBm, and its minimum cycle sleep current is 2uA.

TLSR8258 has great potential to become the wireless microcontroller of choice for future smart furniture, IoT transformation, and industrial automation. Its network characteristics comply with the ZIGBEE 3.0 standard and provides a complete application integration solution based on the IEEE802.15.4 standard ISM band. The product has been inspected and certified by a series of authoritative radio frequency instruments, and combined with many years of market experience and the actual needs of users in this industry, it integrates the extremely complex communication protocols of wireless products into the built-in SoC, supports serial port transparent transmission mode, and integrates quickly and easily. The self-organizing network function provides multiple configurable ADC, IO, and PWM interfaces, simplifying the complexity and greatly simplifying the complex development process of wireless products, allowing your products to be quickly put into the market at a lower cost.

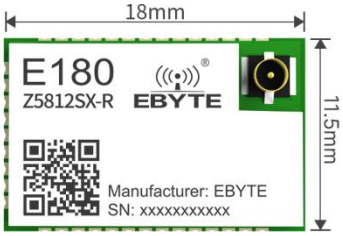



E180-Z5812 SP (terminal)




E180-Z5812 SP-R (router)


E180-Z5812 SX (terminal)




E180-Z5812 SX-R (router)

## 1.2 Functional features

- Centralized network management: ZIGBEE 3.0 security standard centralized network access mechanism, data security and reliability;
- Supports ZCL standard protocol and configurable ZCL protocol specifications, and can simulate the access and control of tens of thousands of smart devices in hundreds of categories under the zigbee protocol. It can be connected to multiple smart home gateways and platforms without the need to customize firmware.
- It can support the access and control of multiple smart home terminal products (coordinator version only), and supports the access and control of hundreds of categories and tens of thousands of smart devices under the Zigbee protocol, without the

need for specially customized firmware.

- Large capacity: 512K capacity flash, 64K capacity RAM, network nodes can be expanded to more than 100;
- Role switching: Users can use serial port commands to switch the device between the two types of terminal and dormant terminal (limited to terminal node version);
- Supports multiple network topologies: point-to-point, star network, MESH network;
- Network self-healing: If the intermediate node of the network is lost, other networks will automatically join or maintain the original network;
- Address search : Users can find the corresponding short address based on the MAC address of the node that has joined the network, and can also find the corresponding long address of each node in the network based on the node's short address.
- Network management: The coordinator version can manage the MAC addresses of all network nodes (routing nodes and terminal nodes), and add and delete nodes freely.
- Data security: Integrated ZIGBEE 3.0 secure communication standard, the network contains multi-level security keys;
- Serial port configuration: The module has built-in serial port instructions. Users can configure (view) the parameters and functions of the module through the serial port instructions.
- Network PAN_ID change: Any switching of network PAN_ID, the user can customize the PAN_ID to join the corresponding network or the PAN_ID will be automatically selected to join the network;
- PWM control: local/remote PWM control, 4 PWM channels for user selection;
- ADC control: local/remote ADC reading, 2 ADC channels for user selection;
- One-click baud rate recovery: If the user forgets or does not know the baud rate, this function can be used to restore the default baud rate to 115200.
- Serial port reception wake-up: supports the serial port reception wake-up function. When the module is in sleep state, it will wake up when it receives a frame of data less than or equal to 10 bytes. This data is a wake-up frame used to wake up the module and will not be treated as data. .
- Module reset: Users can reset the module through serial port commands.
- Restore factory settings: Users can restore the module to factory settings through serial port commands.
- Remote management and remote configuration: Users can use ZCL commands to remotely manage and remotely configure other devices in the network

## 1.3 Introduction to equipment types

There are four logical device types in the ZigBee network: Coordinator (coordinator), Router (router), End-Device (non-sleep terminal) and Sleep-End-Device (sleep terminal). The ZigBee network consists of a Coordinator, multiple Routers and multiple End_Devices (the end nodes can be divided into dormant terminals and non-dormant terminals). This product has three forms, namely dedicated coordinator, dedicated router and dedicated terminal node. The dedicated terminal node supports two device types: End-Device (non-sleep terminal) and Sleep-End-Device (sleep terminal).

Note: The E180-Z5812 series only supports routers and terminals. E180-Z5812 SP and E180-Z5812 SX are terminal modules, and E180-Z5812 SP-R and E180-Z5812 SX-R are router modules.

### 1.3.1 Coordinator

It has the function of establishing and managing the network, controls whether other nodes are allowed to join the network, stores network information, and has all the functions of a routing device. Its main tasks are to manage the network, record sub-node information, and forward messages. At the same time, the coordinator It is necessary to authenticate the terminal permissions requesting network access.

### 1.3.2 Router

Allow other nodes to connect to the routing device to expand the coverage of the network. Its main task is to forward messages, play the role of relay routing, and has
all the functions of terminal devices. If there are multiple paths from one node to another node, and one of the paths fails, the network will automatically adjust to other optimal paths
for transmission to ensure data arrival. The router can create its own network or join someone else's network. The router is always active, so it must be powered by mains power.

### 1.3.3 Non-sleeping terminal

The main task of the terminal device is to send and receive messages, and other nodes are not allowed to connect to the terminal device. The non-sleeping terminal is always working and can receive and send data at any time.

### 1.3.4 Sleep terminal

The dormant terminal enters the dormant state when there is no data to be sent or received, and the dormant current is as low as about 2uA.

When you need to send wireless data or perform command operations, you must first send a wake-up frame through the serial port, with a length of at least 1 byte ("00" is recommended). The wake-up time lasts for 250ms , during which serial port data (HEX commands, valid load), after successfully receiving a frame of serial port data, the device processes the serial port data and then re-enters sleep .

The dormant terminal can also be woken up through the function pin WAKE. WAKE defaults to high level. Pulling the WAKE pin low will cause the module to continuously wake up. Release the WAKE pin to restore the default high level and the module will resume sleep.

When it is necessary to receive data, the data is received through periodic wake-up. The longer the wake-up period is set, the more delayed the reception will be. The wake-up period is set to 4 levels (0~3) through Attribute 0x0004 under cluster 0xFC08 under port 01, corresponding to 1 second, 3.3 seconds, 5 seconds and 1 minute respectively. If you only need to upload data, you can set the wake-up period to be greater than 1 minute to reduce power consumption, such as battery-powered sensors.

## 1.4 Application scenarios

- Smart home and industrial sensors, etc.;
- Security systems, positioning systems;
- Wireless remote control, drone;
- Wireless game remote;
- Healthcare products;
- Wireless voice, wireless headphones;
- Advanced Meter Infrastructure (AMI);
- Automotive industry applications;
- Building automation solutions;
- Agricultural greenhouse automation applications;

# 2 Specifications

## 2.1 Limit parameters

| The main parameters | Performance | | Remark |
|---|---|---|---|
| | Minimum value | Maximum value | |
| Supply voltage (V) | 1.9 | 3.6 | Exceeding 3.6V will permanently burn the module |
| Blocking power (dBm) | - | 10 | Less likely to be burned if used at close range |
| Working temperature (℃) | -40 | +85 | Industrial grade |

## 2.2 Working parameters

| The main parameters | | Performance | | | Remark |
|---|---|---|---|---|---|
| | | Minimum value | Typical value | Maximum value | |
| Working voltage (V) | | 1.9 | 3.3 | 3.6 | ≥3.3V guarantees output power |
| Communication level (V) | | | 3.3 | | Using 5V TTL risks burning out |
| Working temperature (℃) | | -40 | - | +85 | Industrial grade design |
| Working frequency band (MHz) | | 2405 | - | 2480 | Support ISM frequency band |
| Power consumption | Emission current (mA) | | 24 | | 24mA max at 12dbm |
| | Receive current (mA) | | 9 | | |
| | Sleep current (μA) | | 2.5 | | Periodic sleep current averages 2.5uA |
| Maximum TX power (dBm) | | | 12 | | |
| Air data rate (bps) | | | 250k | | |

| The main parameters | Description | Remark |
|---|---|---|
| Reference distance | 200m /500m | Between two points (Zigbee network supports routing multi-hop function, and the transmission distance can be extended by adding routers). |
| Weight | 0.9g | |
| Protocol | ZigBee 3.0 | |
| Package | SMD | |
| Interface mode | 1.27mm | Stamp hole |
| IC full name | TLSR8258F512ET32 | |
| FLASH | 512KB | |

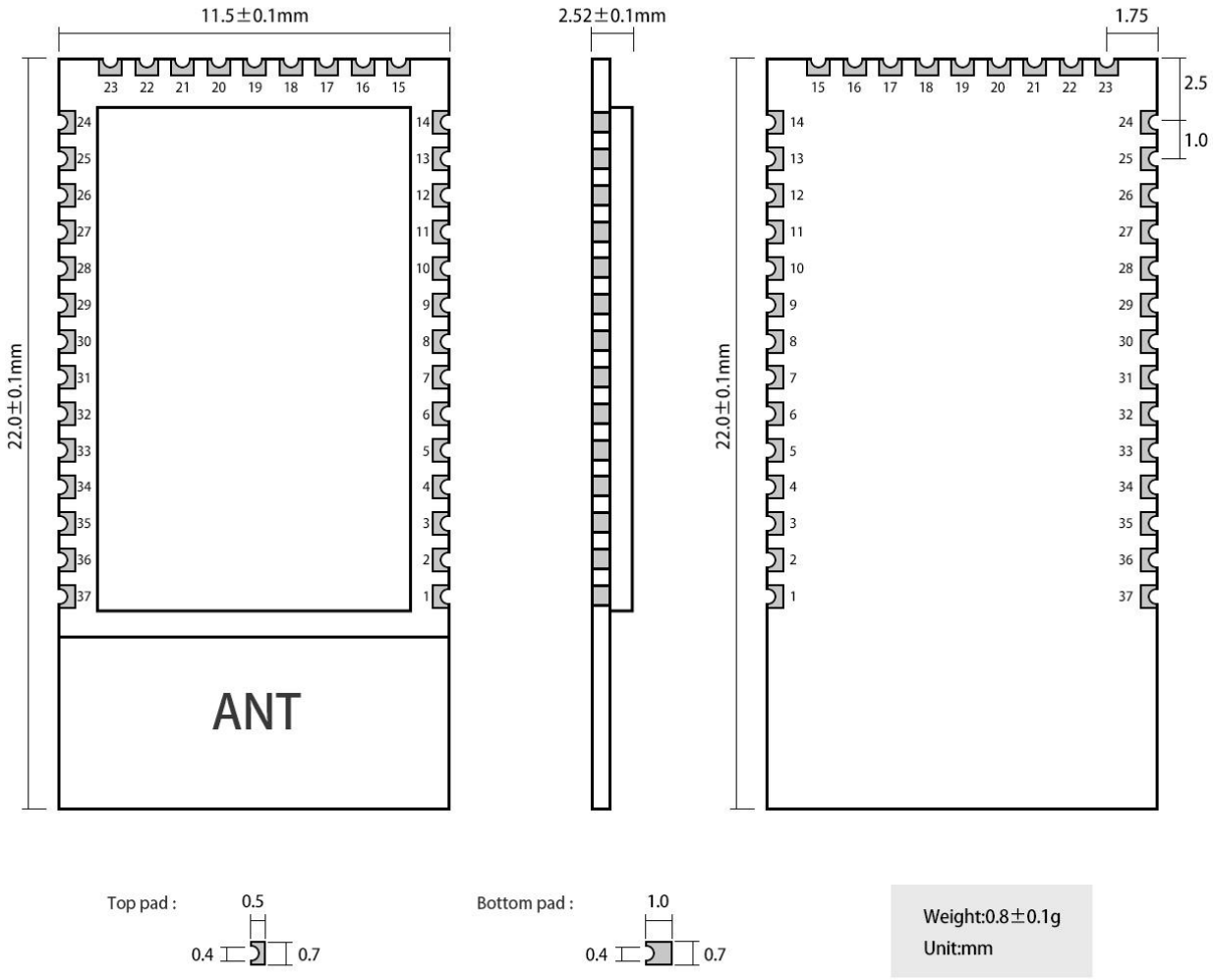
| | | |
|---|---|---|
| RAM | 64KB | |
| Core MCU | 32-bit MCU | |
| Dimensions | 11.5*22mm | E180-Z5812SP, E180-Z5812SP-R dimensions |
| | 1 1 .5 * 18mm | E180-Z5812SX, E180-Z5812SX-R dimensions |
| Antenna interface | PCB | E180-Z5812SP, E180-Z5812SP-R, equivalent impedance is about 50Ω |
| | IPEX | E180-Z5812SX, E180-Z5812SX-R, equivalent impedance is about 50Ω |

## 2.3 Performance parameters

| The main parameters | Description | Remark |
|---|---|---|
| Routing table size | 48 | Router module |
| The maximum number of neighbors of the router | 26 | Router module |
| Maximum number of terminal connections on the router | 16 | Router module |
| Maximum frame length for transparent transmission | 77 bytes | Broadcast or multicast |
| | 240 bytes | Unicast |
| Baud rate ( bps ) | 9600/19200/38400/57600/115200 | |
| Number of PWM output channels | 4 channel | |
| PWM accuracy | 1KHz | |
| PWM minimum gradient period | 10ms | |
| ADC sampling accuracy | 12bit | |
| Number of ADC channels | 3 | 2 external, 1 internal voltage |
| Maximum number of ZCL ports that can be created | 8 | router |
| | 6 | terminal |
| The maximum cumulative number of ZCL clusters that can be created | 32 ( router ) | All ports share all cluster tables |
| | 24 (terminal) | |
| The maximum cumulative number of ZCL attributes that can be created | 128 items (router) | All ports share all attribute spaces |
| | 96 items (terminal) | |
| Maximum ZCL attribute cumulative data space | 1024 Byte (router) | The cumulative size of data pointed to by all attributes under all ports |
| | 768 Byte ( Terminal ) | |

# 3. Mechanical Dimensions and Pin Definitions

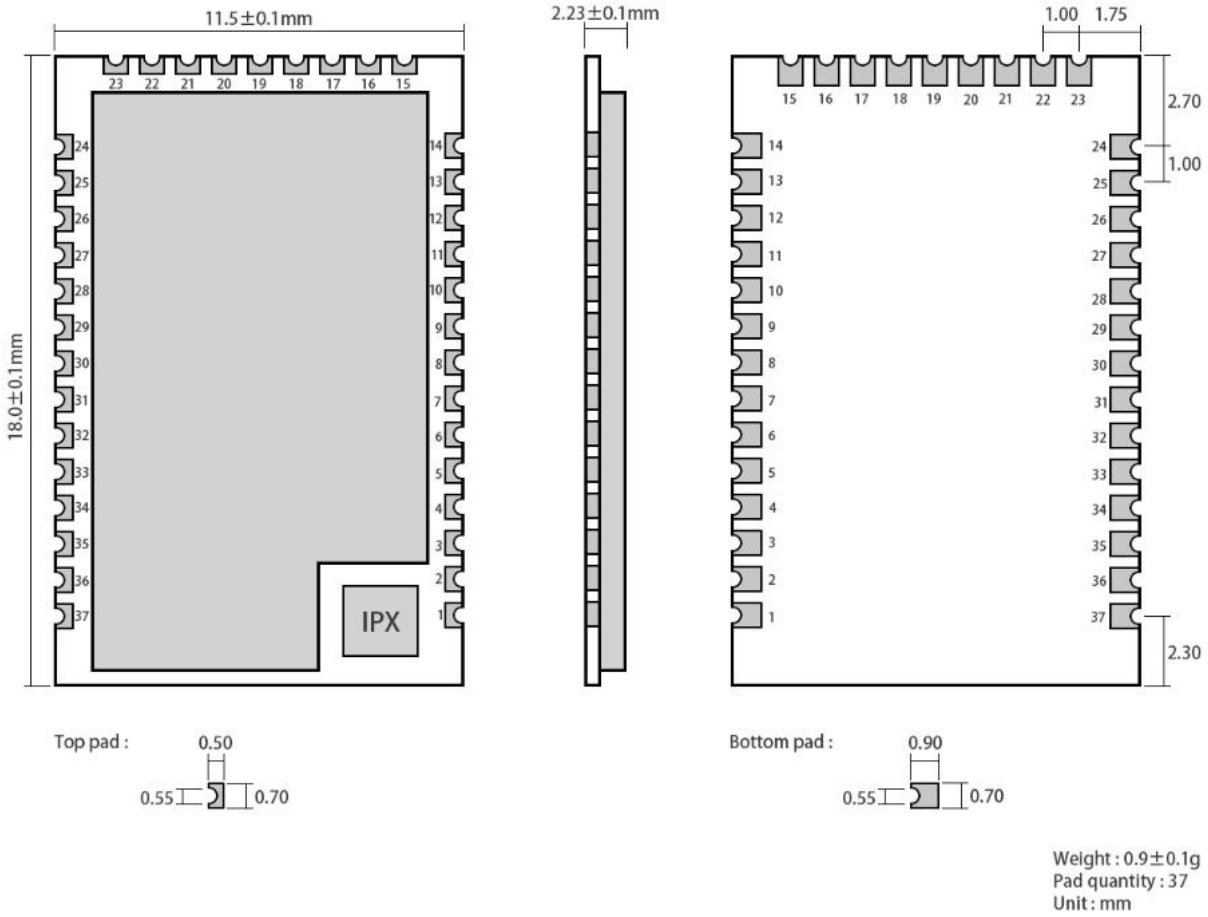
E180-Z5812SP, E180-Z5812SP-R:




| Pin number | Pin name | Pin direction | Pin usage |
|---|---|---|---|
| 1 | NC | - | Reserved, left empty |
| 2 | GND | - | Ground wire, connected to the power reference ground |
| 3 | NC | - | Reserved, left empty |
| 4 | PD3(WAKE) | enter | The WAKE pin is mainly used to wake up dormant terminals. It is high level when powered on. When the pin is pulled low externally, the dormant terminal device will be awakened. |
| 5 | PD7(TX) | output | Serial port transmit port TX |
| 6 | PA0(RX) | enter | Serial receiving port RX |
| 7 | NC | - | Reserved, left empty |

| | | | |
|---|---|---|---|
| 8 | NC | - | Reserved, left empty |
| 9 | PD4(LINK) | enter | Network connection pin. |
| 10 | PA1( BAUD_ R ) | enter | The UART_ BAUD _RESET pin is used to reset the device baud rate. It defaults to high level when powered on. In any mode, if this pin is pulled low for more than 1000ms, the module serial port parameters will return to the default 115200. |
| 11 | PB1(LED1) | output | Reserved LED indication |
| 12 | PC0(GPIO0) | input Output | GPIO input/output port 0 |
| 13 | VCC | - | Module power supply positive reference voltage, voltage range |
| 14 | GND | - | Ground wire, connected to the power reference ground |
| 15 | PB4(GPIO1) | input Output | GPIO input/output port 1 |
| 16 | NC | - | Reserved, left empty |
| 17 | NC | - | Reserved, left empty |
| 18 | PB5(AUX) | output | The AUX pin indicates the current working status of the device. When the pin is low, it indicates that the device is busy. When the pin is high, it indicates that the device is idle. |
| 19 | NC | - | Reserved, left empty |
| 20 | NC | - | Reserved, left empty |
| twenty one | PB6(ADC1) | enter | ADC detection port 1 |
| twenty two | PB7(ADC2) | enter | ADC detection port 2 |
| twenty three | NC | - | Reserved, left empty |
| twenty four | NC | - | Reserved, left empty |
| 25 | SWCLK | input Output | Serial debug interface, serial line clock |
| 26 | SWDIO | input Output | Serial debugging interface, serial data input and output |
| 27 | PC2(PWM0) | output | PWM output port 0 |
| 28 | PC3(PWM1) | output | PWM output port 1 |
| 29 | PC4(PWM2) | output | PWM output port 2 |
| 30 | PC1(NET) | output | The NET pin indicates the current network status of the module. Flashing indicates that the network is being configured or bound. |
| 31 | NC | - | Reserved, left empty |
| 32 | PD2(PWM3) | output | PWM output port 3 |
| 33 | NC | - | Reserved, left empty |
| 34 | NC | - | Reserved, left empty |
| 35 | NC | - | Reserved, left empty |
| 36 | GND | input Output | Ground wire, connected to the power reference ground |
| 37 | nRESET | enter | reset pin |

E180-Z5812SX, E180-Z5812SX-R:




| Pin number | Pin name | Pin direction | Pin usage |
|---|---|---|---|
| 1 | NC | - | Reserved, left empty |
| 2 | GND | - | Ground wire, connected to the power reference ground |
| 3 | NC | - | Reserved, left empty |
| 4 | P D3(WAKE) | enter | The WAKE pin is mainly used to wake up dormant terminals. It is high level when powered on. When the pin is pulled low externally, the dormant terminal device will be awakened. |
| 5 | PD7 ( TX ) | output | Serial port transmit port TX |
| 6 | PA0 ( RX ) | enter | Serial receiving port RX |
| 7 | NC | - | Reserved, left empty |
| 8 | NC | - | Reserved, left empty |
| 9 | PD 4 (MODE) | enter | Working mode switching pin, the working mode switches when the pull-down time is greater than 500ms. |
| 10 | P A1 ( BAUD_R ) | enter | The UART_BAUD_RESET pin is used to reset the device baud rate. It defaults to high level when powered on. In any mode, if this pin is pulled low for more than 1000ms, the module serial port parameters will return to the default 115200. |
| 11 | P B1(ACK) | output | The ACK pin is used to indicate the last user data transmission status. The pin is pulled low before starting the transmission. After the transmission is |

| | | | |
|---|---|---|---|
| | | | successful, the pin is pulled high. |
| 12 | P C0 (GPIO0) | input Output | GPIO input/output port 0 |
| 13 | VCC | - | Module power supply positive reference voltage, voltage range |
| 14 | GND | - | Ground wire, connected to the power reference ground |
| 15 | P B4 (GPIO1) | input Output | GPIO input/output port 1 |
| 16 | NC | - | Reserved, left empty |
| 17 | NC | - | Reserved, left empty |
| 18 | P B5(AUX) | output | The AUX pin indicates the current working status of the device. When the pin is low, it indicates that the device is busy. When the pin is high, it indicates that the device is idle. |
| 19 | NC | - | Reserved, left empty |
| 20 | NC | - | Reserved, left empty |
| twenty one | P B 6 (ADC1) | enter | ADC detection port 1 |
| twenty two | P B 7 (ADC2) | enter | ADC detection port 2 |
| twenty three | NC | - | Reserved, left empty |
| twenty four | NC | - | Reserved, left empty |
| 25 | SWCLK | input Output | Serial debug interface, serial line clock |
| 26 | SW DIO | input Output | Serial debugging interface, serial data input and output |
| 27 | P C2 (PWM0) | output | PWM output port 0 |
| 28 | P C3 (PWM2) | output | PWM output port 2 |
| 29 | P C 4 (PWM3) | output | PWM output port 3 |
| 30 | PC1 (LINK) | output | The LINK pin indicates the current network status of the module, and the output high level meter has been added to the network. |
| 31 | NC | - | Reserved, left empty |
| 32 | P D2 (PWM1) | output | PWM output port 1 |
| 33 | NC | - | Reserved, left empty |
| 34 | NC | - | Reserved, left empty |
| 35 | NC | - | Reserved, left empty |
| 36 | GND | input Output | Ground wire, connected to the power reference ground |
| 37 | n RESET | enter | reset pin |

# 4 Serial Port Input and Output

There are 4 operating modes for serial port input and output frames: HEX command mode (configuration mode), universal mode, transparent transmission mode, and AT command mode.

Note: E180-Z5812SP and E180-Z5812SP-R are not compatible with the old version E180-Z5812SP, and E180-Z5812SX and E180-Z5812SX-R are not compatible with the old version E180-Z5812SX. The module is connected to the serial debugging assistant, and the module is reset or powered back on. The serial debugging assistant prints data similar to 55 0D 80 00 00 10 E5 7E 40 E0 C3 38 C1 A4 35 for the new version module, and does not print data for the old version module.

## 4.1 HEX instruction mode

The factory default of the module is HEX command mode. In this mode, only data frames in HEX command format can be input and output.

### 4.1.1 HEX command frame format

The HEX instruction format is a fixed pattern of "frame header + frame length + frame load". The instruction frame length is variable. Instruction input is not affected by instruction packet sticking, and the input instruction has a timeout protection mechanism, which effectively solves the problem of instruction packet breakage. Each input command has corresponding command feedback to confirm whether the module is working properly and whether the command is executed correctly. The HEX command mode is full-duplex mode. When the module status changes or data is received, the corresponding HEX command is output in real time through the UART_TX port.

For details on the format and analysis of HEX, please refer to the document "Ebyte ZigBee3.0 Module HEX Command Standard Specification". This document focuses on the characteristics of E180-ZG120 in HEX command mode.

**HEX instruction format:**

| Frame header (1 byte) | Frame length (1 byte) | **Frame payload (variable length 3~255 bytes)** | | | |
|---|---|---|---|---|---|
| | | Command type (1 byte) | command code (1 byte) | command data (variable length 0~252 bytes) | XOR check (1 byte) |

**Frame header:** Hexadecimal fixed byte 0x55

**Frame length:** 1 byte length, value range 3~255 (hexadecimal: 0x03~0xFF)

**Frame payload:** The frame payload includes command type, command code, command data and XOR check. The length is determined by the frame length.

**Command type:** Classify the command according to its mode and working mechanism.

**Command code:** The code corresponding to the command, 1 byte in length, each command has a unique command code.

**Command data:** Parameters attached to the command execution, minimum 0 bytes, maximum 252 bytes.

**XOR check:** XOR8 checksum of the entire command payload (command type, command encoding, command data).

### 4.1.2 HEX instruction classification

**HEX instructions can be divided into three categories based on input and output methods :**

- Input command: The host computer inputs the command of the module, which can be used to configure the module or send it

wirelessly. The command type of the input command is less than 0x0F.

- Feedback command: After the module receives and executes the command from the host computer, it feeds back the execution result to the host computer. The command type and command code of the feedback command are the same as the input command.
- Asynchronous command: A command actively sent to the host computer during the operation of the module. This command corresponds to asynchronous events in ZigBee applications. The command type of asynchronous commands is greater than 0x80.

**HEX instructions can be further subdivided into the following seven types :**

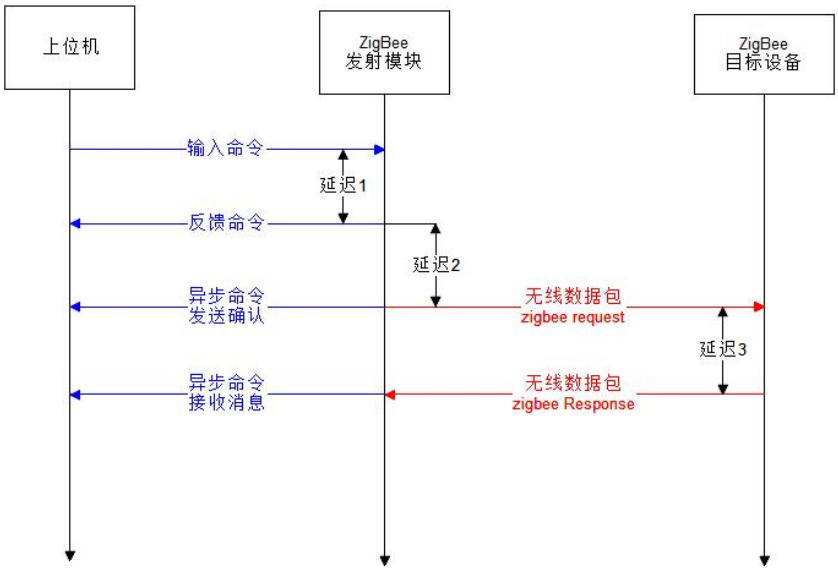
**Enter commands (including feedback commands)**

- Local configuration command: Command type 0x00, used for local settings of the module.
- Network management command: Command type 0x01, used to manage the network layer of other modules during networking.
- ZCL sends command: command type 0x02, used by the module to control other modules or third-party devices, in compliance with ZCL specifications.

**Asynchronous commands:**

- System notification command: command type 0x80, module status change notification.
- Network management return command: command type 0x81, other modules or devices receive the gateway management command return message.
- ZCL receive command: command type 0x82, the module receives ZCL layer messages or return messages from other modules or devices.
- Send confirmation: Command type 0x8F, used to diagnose whether the sending of network management commands and ZCL send commands is abnormal.

**Note:** The "Send Confirm" command (the host computer) inputs network management commands and ZCL sending commands to the module and receives the corresponding feedback commands. The commands are not immediately converted into wireless signals and sent out, but are avoided in the same network. Other devices then transmit signals into the air at a rate of 250kbps. The "send confirmation" command is the result of sending the wireless signal. There are only two states for sending confirmation on the module: 0x00 = sending successfully, other values = sending failed. The final correctness of the network management command is judged based on the corresponding network management return command received; the final correctness of the ZCL sent command is also judged based on the corresponding ZCL receive command received. Sending acknowledgment can be used to end waiting for return messages early, and can be used to diagnose other modules or devices for exceptions, instead of wasting limited network resources on meaningless device nodes.




When the module sends network management commands and ZCL send commands by broadcast, due to the impact of broadcast flooding, the send confirmation will be triggered 1 second after the feedback command. Therefore, it is recommended that the sending interval be greater than 1 second when using this module for broadcast or multicast.

## 4.1.3 HEX commands supported by this product

EBYTE's full range of Zigbee products use unified HEX commands. However, due to differences in chip solutions and software configurations, the actual HEX commands supported are different. This section shows the HEX command directory supported by this product. For the format and content of HEX commands, please refer to "EBYTE" ZigBee3.0 module HEX command standard specification》

| Command name | Command type | command code |
|---|---|---|
| local configuration class | | |
| Query the current status of the module | 0x00 | 0x00 |
| Start configuring network | 0x00 | 0x02 |
| Stop distribution network | 0x00 | 0x03 |
| Reset/factory restore | 0x00 | 0x04 |
| Set the local node type (terminal nodes only) | 0x00 | 0x05 |
| Query and set up channels | 0x00 | 0x06 |
| View local group additions | 0x00 | 0x09 |
| Add local group | 0x00 | 0x0A |
| This machine leaves the group | 0x00 | 0x0B |
| Set and query the current transmit power | 0x00 | 0x0D |
| Read local properties | 0x00 | 0x10 |
| Set local properties | 0x00 | 0x11 |
| Automatically bind target | 0x00 | 0x14 |
| Read the network node address table | 0x00 | 0x22 |
| Retransmit device information | 0x00 | 0x28 |

| | | |
|---|---|---|
| notification | | |
| Create ZCL port | 0x00 | 0x40 |
| Add properties | 0x00 | 0x41 |
| Save ports and properties | 0x00 | 0x42 |
| Read and write properties | 0x00 | 0x43 |
| Clear ports and properties | 0x00 | 0x44 |
| Set current scene data | 0x00 | 0x45 |
| Query and set automatic attribute reporting rules | 0x00 | 0x46 |
| View created port information | 0x00 | 0x47 |
| View the list of added properties | 0x00 | 0x48 |
| Edit and modify automatic binding | 0x00 | 0x4C |
| View the module's own binding objects | 0x00 | 0x4D |
| Delete bound object | 0x00 | 0x4E |
| Send ZCL control commands to all bound targets | 0x00 | 0x4F |
| System notification class | | |
| Module start | 0x80 | 0x00 |
| Network status changes | 0x80 | 0x01 |
| Turn network notifications on and off | 0x80 | 0x02 |
| Detect node access to the network | 0x80 | 0x03 |
| Node short address notification | 0x80 | 0x04 |
| Device information notification | 0x80 | 0x05 |
| Node off-grid notification | 0x80 | 0x06 |
| Property remote modification notification | 0x80 | 0x40 |
| Identify notification | 0x80 | 0x41 |
| Scenario execution notification | 0x80 | 0x43 |
| ZDO network management command/network management return | | |
| Query node short address/return | 0x01/0x81 | 0x00 |
| Query node MAC address/return | 0x01/0x81 | 0x01 |
| Query node port information/return | 0x01/0x81 | 0x04 |
| Query node port number/return | 0x01/0x81 | 0x05 |
| Set node constant connection binding/return | 0x01/0x81 | 0x21 |
| Unbind/return node constant connection | 0x01/0x81 | 0x22 |
| View node constant connection binding/return | 0x01/0x81 | 0x33 |
| Delete node/return | 0x01/0x81 | 0x34 |
| ZCL command and ZCL return | | |
| Read device properties/return | 0x02/0x82 | 0x00 |
| Modify device properties/return | 0x02/0x82 | 0x01 |

| | | |
|---|---|---|
| Query attribute reporting rules/returns | 0x02/0x82 | 0x02 |
| Modify attribute reporting rules/return | 0x02/0x82 | 0x03 |
| View all properties/return | 0x02/0x82 | 0x04 |
| View all status with extended fields/return | 0x02/0x82 | 0x05 |
| Receive attribute report proactively | 0x82 | 0x0A |
| Default return frame | 0x82 | 0x0B |
| Send control command | 0x02 | 0x0F |
| receive control command | 0x82 | 0x0F |

- E180-Z5812 supports ADC. Use the "read local attributes" command to read attributes 0x0100 and 0x0101. These two attributes are the ADC values of ADC1 and ADC2 respectively. You can also remotely access the ADC through the ZCL command to read cluster 0xFC08. Attributes 0x0100 and 0x0101 under (need to use manufacturer code 0x2000).

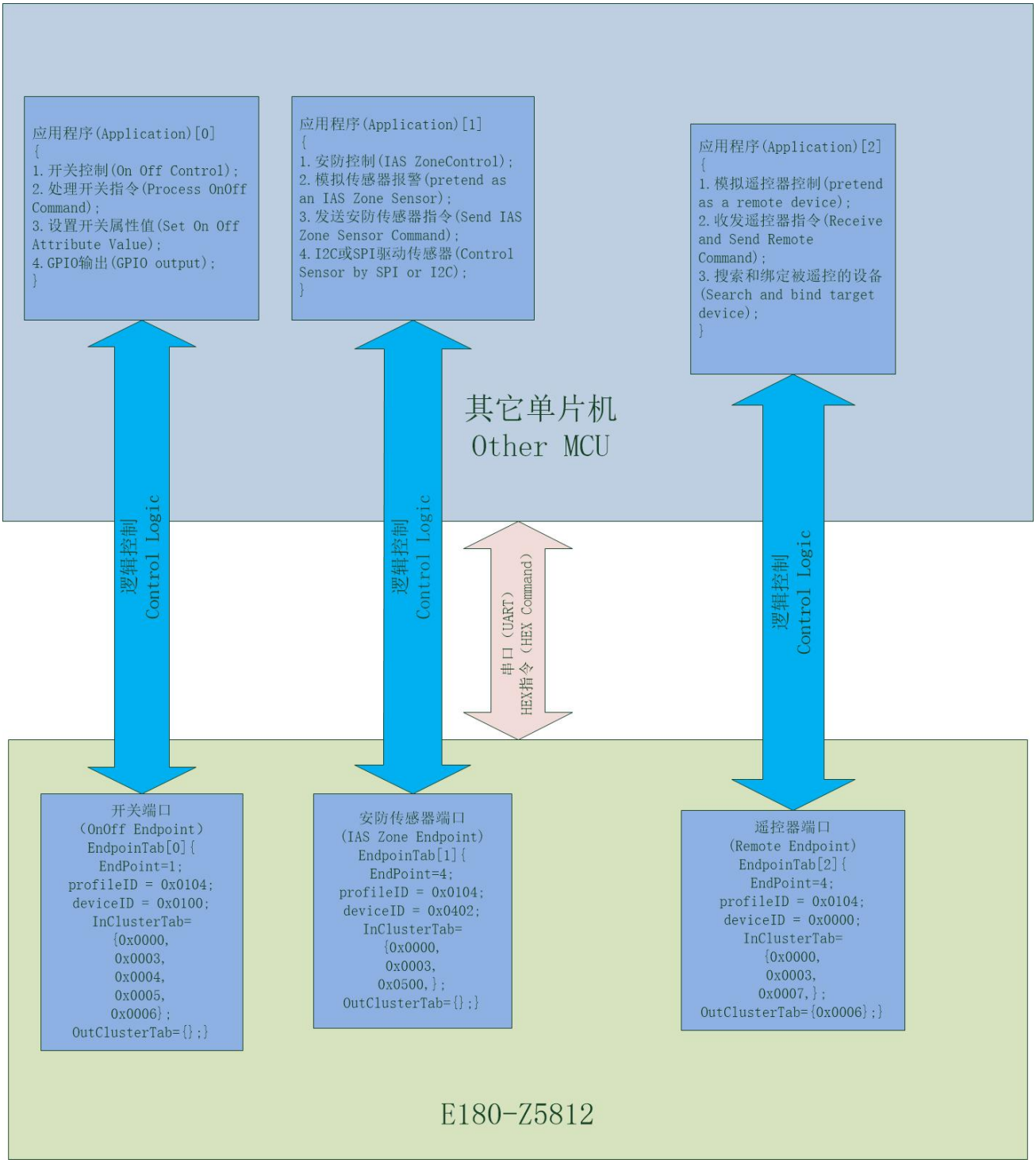
## 4.2 Universal mode

Universal mode is a new function in HEX command mode. In this mode, the module can be configured according to ZCL standard specifications and simulate various smart devices that comply with ZCL specifications. The configured module in universal mode can be correctly identified as the corresponding access device when connected to a third-party smart gateway (such as a Zigbee gateway), and can send and receive control instructions for the gateway to interact with the device.

### 4.2.1 Precautions for universal mode

- In the universal mode, the node must be configured when not connected to the network in order to simulate the access of various Zigbee smart devices and the sending and receiving of control instructions.
- When using universal mode, you must first use the "Create ZCL Port" command to create a device port, then use "Add Properties" to add all properties under this port, and then use "Save Port and Properties" to save the properties created for this port.
- When adding attributes, the attribute ID must be added from small to large first, and then the cluster ID must be added from small to large. After all the attributes under a cluster are filled up, the attributes of the next cluster will be added.
- When creating a port, you must fill in the cumulative attributes of all clusters under the port. If the actual added attributes are greater than this value, the addition will fail.
- When adding attributes, the actual attribute clusters added can be more than the clusters planned to be added when creating the port. These clusters will not be discovered by the gateway after the device is connected to the gateway, but if the gateway is forced to read the attributes under these clusters, it will be valid. numerical value. (can also be called hidden cluster)
- In the coordinator and routing node modes, the cumulative number of added ports does not exceed 8, the cumulative number of added attributes for 8 ports does not exceed 32 clusters, and the cumulative number of added attributes does not exceed 128 items.
- In terminal node and dormant node modes, the cumulative number of added ports does not exceed 6, the cumulative number of added attributes for 6 ports does not exceed 24 clusters, and the cumulative number of attributes added does not exceed 96 items.
- You need to modify the ZCL attribute configuration information, or fall back to the transparent transmission module mode, and directly use "Clear Ports and Attributes". This command must be exited from the network or not connected to the network to take effect.

## 4.2.2 Universal mode schematic block diagram




- The external MCU interacts with the E180-Z5812 module entirely through HEX commands. The external MCU can simulate different device access and control through different HEX.
- This block diagram is just a schematic diagram of the software architecture. Three ports are created to represent three different devices. In fact, according to the processing capabilities of the external MCU, a smaller number of ports can be created, or multiple ports with the same function can be created to reduce the load on the external MCU. processing pressure.
- The initialization information of port creation and attribute values is stored in the FLASH of E180-Z5812, so you can use PC

serial port instructions to create ports and initialize attribute values on the zigbee module, and then the module hardware is connected to the MCU, and the MCU application only processes the application layer of command interaction and device control, eliminating the need to design, create ports and initialize properties.

### 4.2.3 Universal mode usage tutorial

Note: See Chapter 6 for instructions on how to use universal mode.

## 4.3 Transparent transmission mode

In data transparent transmission mode, any data input to the serial port will be sent out through wireless signals. The module that receives the transparent transmission data directly prints the data frame in transparent transmission mode or AT command mode. For example, the receiving end is in HEX command mode. Then the frame data is output in ZCL command format.

### 4.3.1 ZCL specification for data transparent transmission

Data transparent transmission complies with ZCL specifications, and the specifications are as follows. Any Zigbee device can send and receive data input and output from the E180-Z5812 serial port as long as it is set according to the following specifications.

- port=1
- Profile=0x0104
- cluster=0xFC08
- manufacture code=0x2000
- Command type: Special Command
- Command direction: Server to Client
- Command ID: 0x00

Note: When the coordinator or other nodes are in HEX mode and send data to the E180ZG120 module in transparent transmission mode, they also need to follow the ZCL specification, that is, use cluster=0xFC08 cluster, Manufacturing Code=0x2000, and change the command direction to Client to Server. The command ID is 0x00.

### 4.3.2 Target setting of data transparent transmission

The data transparent transmission target sets the module's local attribute DstAddr and local attribute DstEP. These two attributes are located at port 1 of the module, cluster=0xFC08. The attribute IDs are 0x0001 and 0x0002 respectively, and the data types are UINT16 and UINT8 respectively.

DstAddr is the short address of transparent transmission reception, and DstEP is the target port of transparent transmission reception. Transparent transmission has four modes: on-demand transmission, broadcast transmission, multicast transmission, and binding transmission. When DstEP is set to other values, it is used to reserve transparent transmission for ZigBee modules with multiple serial ports.

| Transparent transmission mode | DstAddr | oeLh |
|---|---|---|
| On demand (to main serial port) | Short address of the other party | 1 |
| On demand (to serial | Short address of the other | 2 |

| | | |
|---|---|---|
| port 2)<br>Note: Reserved dual serial port function | party | |
| broadcast | 0xFFFF | 0xFF |
| multicast | 16bit group address | 0 |
| Bind to send | 0xFFFE | 0xFE |

### 4.3.3 Binding transparent transmission target

In the binding transparent transmission mode, the module can find the transparent transmission target through the MAC address to cope with changes in the short address of the transparent transmission target. There are three ways to set binding:

a) The coordinator assigns a transparent transmission target to the module through the HEX command "Set node constant connection binding ("HEX command")". If the module knows the other party's MAC address, it can also Send this command to yourself in HEX command mode.

b) Both modules are in HEX command mode and send the local configuration command "Automatically bind target (command code 0x14)". The transparent transmission target module sends first, wait for about 1 to 3 seconds before the LINK indicator light flashes, and then the transparent transmission source module sends the command.

c) In AT command mode, the two modes use the "AT+FIND" command to bind each other. The operation method is the same as the local configuration command "Automatically bind target (command code 0x14)".

d) In any mode, input a key signal (falling edge 20ms~200ms) on the PD4 pin of the two modules that need to be bound to each other. The operation method is the same as the local configuration command "Auto-binding target (command code 0x14)".

**Feedback information for data transparent transmission**

When the E180ZG120 module sends data in transparent transmission mode, there will be result feedback. The feedback information is less than or equal to 4 bytes. In order to distinguish the feedback information from the received transparent transmission data, it is recommended that the transparent transmission data be greater than or equal to 5 bytes. Feedback information is as follows

"OK": sent successfully

"FAIL": Failed to send

"ERRO": sending error, such as the sending buffer is full, the module is not connected to the network, and the module is offline

"OFF": The module is offline. This phenomenon will occur on terminal nodes and dormant terminals.

"NET": When the module comes back online, this phenomenon will occur on terminal nodes and dormant terminals.

"BUSY": During transparent transmission, the previous packet of data has not been transmitted completely and the next packet of data continues to be transmitted, which will cause data congestion.

## 4.4 AT command mode

AT mode is a special state in transparent transmission mode and is used for module configuration in transparent transmission mode. AT commands are in ASCII string format, which is convenient for manual direct input and mnemonic. AT commands adopt the format of "AT+command code", and the command code is a fixed string. For detailed analysis of AT commands, please refer to "Ebyte ZigBee 3.0 Module AT Command Standard Specification". There are three input forms of AT commands: execution type, query type, and setting type.

### 4.4.1 Execution type

The format of the execution command is the direct format of "AT+command code". The execution command ends with the last byte of the command code, and no bytes are continued after it, including the carriage return symbol, otherwise the input is invalid. Such as "AT+JOIN", "AT+LEAVE". Executing the command input effectively returns "OK\r\n", that is, receiving an "OK" with a carriage character at the end. If the input command ends incorrectly, the module returns "INVALID\r\n".

## 4.4.2 Query formula

The format of the query command is "AT+command code?", which ends with an ASCII "?" (hexadecimal 0x3F). The query command is used to query the current value of a parameter of the module and print the query value through the serial port in ASCII format.

## 4.4.3 Setting type

The format of the setting command is "AT+command code=numeric value", that is, the end of the command code needs to be followed by the ASCII "=", and the "=" is followed by the numerical value. Depending on the command, the numerical value can be input in decimal or hexadecimal format, expressed in the format of %d or %x. If multiple parameters are entered, they need to be separated by ",". For details, please refer to "Ebyte ZigBee 3.0 Module AT Command Standard Specification".

## 4.4.4 AT command directory

| Command function | command code | implement | Inquire | set up |
|---|---|---|---|---|
| Exit AT mode to HEX mode | AT+EXIT | Y | N | N |
| Network or create a new network | AT+JOIN | Y | N | N |
| Module reset | AT+RESET | Y | N | N |
| Off the grid | AT+LEAVE | Y | N | N |
| Enter transparent transmission mode | AT+SEND | Y | N | N |
| Automatically bind target | AT+FIND | Y | N | N |
| Read device information | AT+INFO | Y | N | N |
| Set or read baud rate | AT+BAUD | N | Y | Y |
| Set or read the target address | AT+DSTADDR | N | Y | Y |
| Set or read the target port | AT+DSTEP | N | Y | Y |
| Set or read sleep level | AT+LPLEVEL | Y | Y | Y |
| Set or read ModbusID | AT+MBID | Y | Y | Y |
| PWM output | AT+PWM | N | N | Y |
| Acquire ADC | AT+ADC | N | N | Y |
| Enter test mode | AT+TEST | N | N | Y |

Note: The red fonts in the above table are special support instructions for E180-Z5812 series modules .

## 4. 4.5 New AT command

① **Set or read Modbus ID**

Operation 1, set Modbus ID

Enter the command: "AT+MBID=%d", %d is an integer from 0 to 255

Return command: "OK\r\n"

Operation 2, check Modbus ID

Enter the command: "AT+MBID?", %d is an integer from 0 to 255

Return command: "MBID=%d\r\n"

**② Set PWM output**

Set PWM output

Enter the command: "AT+PWM=%d,%d", the first %d is the PWM channel, the value range is 0~3, the second %d is the PWM pulse width, the value range is 0~1000.

Return command: "PWMCH=%d, PWM=%d\r\n", the first %d is the PWM channel, and the second %d is the PWM pulse width.

**③ Set up ADC acquisition**

Collect ADC readings

Enter the command: "AT+ADC=%d", %d is the ADC channel, the value range is 0~2

Return command: "CH:%d ADC=%d\r\n", the two %d are ADC channel and ADC reading respectively.

**④ Enter test mode**

Enter the command: "AT+TEST=%d", %d is the test channel, and 11~26 are valid values.

Return command: "OK\r\n"

After using this command, the module will continuously output beacon request frames, and the signal can be checked on the spectrum analyzer.

# 4.5 Mode switching

## 4.5.1 Mode switching table

| | | target mode | | | |
|---|---|---|---|---|---|
| | | HEX command | Universal mode | AT command | Transparent transmission mode |
| Current mode | HEX command | | Configuration commands "Create ZCL port", "Add attributes", "Save port and attributes" (see "HEX Commands" 2.1.28 to 2.1.30) | Configuration command "Set local attributes ("HEX command 2.1.16")"<br>----------<br>Set attribute 0x0003 to a non-zero value | Configuration command "Set local attributes ("HEX command 2.1.16")"<br>----------<br>Set attribute 0x0003 to a non-zero value |
| | AT command | "AT+EXIT" | | | "AT+SEND" |

| | | | | | |
|---|---|---|---|---|---|
| | Transparent transmission mode | Send 3 characters "+++" | | Send 3 characters "+AT" | |
| | Universal mode | After exiting the network, use the configuration command "Clear Ports and Properties"<br>Directly restore factory settings | | none | none |

## 4.5.2 Precautions for mode switching

- The module actually only has HEX command mode and transparent transmission mode. AT command mode can be regarded as a special state in transparent transmission mode.
- Restart the module in AT command mode. The default state of the module is transparent transmission mode.
- The HEX command mode switches between transparent transmission mode or AT mode, which is determined by the mode before the last switch to HEX mode.
- In universal mode, only HEX commands can be used for control. In universal mode, HEX commands can also be used to simulate data transparent transmission.

# 5 Module Network Distribution and Data Transmission

## 5.1 Module network distribution

### 5.1.1 HEX command network configuration

Modules that are not configured with a network cannot transmit data, and modules that are configured with a network are always in the network with a network, unless the network is removed.

When configuring the network, first let the coordinator configure the network. The newly shipped coordinator does not have a network configured. Use the HEX command "Start Network Configuration ("HEX Command" 2.1.3)" and the coordinator will create a new network. The coordinator will automatically generate a valid PANID to indicate that the network is successfully established, and will open a channel to run the network. The coordinator continues to run on the network after restarting.

If routers, terminal nodes, and dormant terminals need to join the network, the coordinator needs to execute the "Start Network Configuration" command again, indicating that the coordinator accepts new devices into the network. Routers, terminal nodes, and dormant terminals that need to access the network also execute the "Start Network Configuration" command. After the network configuration is completed, routers, terminal nodes, and dormant terminals will receive the "Network Status Change Notification" command regardless of whether the network configuration is successful or not. The "Network Status Change Notification" command will inform the host computer user whether the network configuration is successful. You can also use the "Query Module Current Status" command to check whether the module has been configured for the network.

### 5.1.2 AT command network distribution

When the module is in transparent transmission mode, you can enable AT commands for network configuration. In transparent transmission mode, enter the 3 characters "+AT" to enable AT mode.

When the coordinator is in AT mode, enter "AT+JOIN" to open the network. Enter "AT+JOIN" on the router and end node to open the network.

### 5.1.3 Button configuration

Pin PD4 (network connection pin) inputs a low-level signal lasting 10ms~200ms to trigger the button function. The router or terminal node actively connects to the coordinator when the network is not configured. The coordinator inputs the button signal on this pin. Enter distribution network mode.

It is necessary to pay attention to the routers and terminal nodes that have been configured with the network. If multiple such devices press the network configuration button at the same time, the binding mode will be triggered, that is, each other will be set as the target device for transparent transmission of data.

## 5.2 Data transmission

Whether the data transmission module is in HEX command mode or data transparent transmission mode, the data transmission

methods are only broadcast and on-demand. Broadcast mode includes multicast transmission, and on-demand mode includes binding transmission.

### 5.2.1 Broadcast mode

In this mode, the target short addresses are 0xFFFC (only to the coordinator or router), 0xFFFD (including non-dormant terminal nodes), and 0xFFFF (including all devices of dormant terminal nodes). The destination port is 0xFF.

### 5.2.2 On-demand mode

In this mode, you can accurately control who the data is transmitted to, or precisely control the PWM output of a certain module. In this mode, the short address is the short address of the module that requires precise control, and the target port is the port number of the module that requires precise control (4 PWM output ports correspond to 2, 3, 4, and 5)

### 5.2.3 Multicast mode

In this mode, broadcast control can be performed on some modules that need to be controlled. The destination address is the group number that needs to be controlled, and the destination port is 0. Using this function requires grouping the target modules in advance, and the grouping operation needs to be accurate to the port of the module. If you only need to control the input and output of data transparent transmission, you need to add port 1 of the module to the desired group. If you need to control PWM output in groups, you need to add the corresponding PWM port of the module to the group. When controlling PWM in groups, multiple different PWM serial numbers of multiple different modules can be added to the designated group, and the desired control effect can be achieved when controlling them in multicast.

### 5.2.3 Binding transfer mode

The binding transmission mode is a polling-on-demand method that achieves addressing by locking the MAC address of the target device without needing to remember the other party's short address. E180-Z5812 series modules only use bonded transmission in two situations. A node can bind multiple different target addresses at the same time.

- Binding transparent transmission: In transparent transmission mode, the target address is 0xFFFE, the target port is 0xFF, and the transparent transmission data is on-demand and transmitted to the latest bound target device in the binding table.
- Status reporting: After setting up the binding, E180-Z5812 will report the serial port mode, Modbus ID, PWM on-off status, and PWM pulse width to the target device. For example, the coordinator will automatically set the router or terminal when it detects that the router or terminal is connected. The terminal is bound to the coordinator itself, and the router or terminal will report the serial port mode, Modbus ID, PWM on-off status, and PWM pulse width to the coordinator at a fixed period. If the above status changes, it will also be reported to the coordinator. At the same time, these statuses can also be reported to other routers or terminals, making it easier to obtain the other party's status when using the router or terminal to control the application of the router or terminal.

## 5.3 Set up binding

### 5.3.1 Coordinator remote configuration

Set the binding source device and target device according to " Setting Node Constant Connection Binding (3.3.6) " in the "HEX Command" manual . Note that the binding objects are all ports (virtual devices) on each node, that is, each port needs a unique SN number composed of port number + MAC address.


Note: Each node has several virtual devices that can be used as binding objects.

Using the Ebyte PC tool, you can easily set the binding source and binding target. When configuring binding remotely, in addition to operating on the coordinator, you can also operate on the router or end node. In addition, if the binding target SN is all 0, it means that the binding target is yourself. This method allows the router or terminal node to serve as the master and allow other nodes to bind to it, making it easier to obtain the other party's status.

## 5.3.2 Set binding with one click

This method can only bind transparent transmission targets and cannot report binding status.

After the router or terminal is configured with the network, input a 50ms~500ms low-level signal to the LINK button (PD4) to trigger the button. Two routers or terminal modules trigger the LINK button one after another. The NET pin (PC1) of the module that is triggered first outputs a high and low level of 1Hz, and the second module triggers the button.

One-key binding can also be triggered by using the HEX command " Automatically bind target (2.1.17) " or the AT command "AT+FIND" instead of pressing a button.

## 5.3.3 View bindings

You can use the coordinator to view the binding of the node, and use the HEX command " View node constant connection binding (3.3.8) " to view the binding table. By entering the node short address, you can see which virtual SNs of other devices are bound to all virtual SNs under the node.

Using Ebyte PC software, you can also view other devices bound to each device.


Note: All bindings can be viewed by clicking "View Constant Connections" in the host computer software.

# 6 Universal Mode Usage Tutorial

In universal mode, you need to prepare a zigbee smart gateway (you can choose a crow, a Stanley, a micro, etc.), and use the APP supporting the gateway to add the E180-Z5812 series module.

## 6.1 Routine 1: Simulate a three-way switch to access the gateway and use APP to control the switch

### 6.1.1 Create the first switch port and add properties

The index number of the first port is 0 (use an array to represent the port list), and the port number can be assigned arbitrarily from 1 to 240. However, for better compatibility between the gateway and the cloud platform, it is recommended to use 0x01 for the first port number.

**Enter the "Create ZCL Port" command:**

55 15 00 40 [15] [01] [04 01] [00 01] [05] { [00 00] [03 00] [04 00] [05 00] [06 00] } [00] 51

The contents in "[ ]" are as follows

- The switch has a total of 0x15=21 attributes
- Create port number = 0x01
- Contour ID = 0x0104
- Device ID = 0x0100
- Input number of clusters = 0x05
- Input cluster {0x0000, 0x0003, 0x0004, 0x0005, 0x0006}
- Number of output clusters = 0x00

Return command:

55 16 00 40 [00] [00] [01] [04 01] [00 01] [05] {[00 00] [03 00] [04 00] [05 00] [06 00]} [00] 44

The contents in "[ ]" are as follows

- execution succeed
- Module assigned port index = 0
- Create port number = 0x01
- Contour ID = 0x0104
- Device ID = 0x0100
- Input number of clusters = 0x05
- Input cluster {0x0000, 0x0003, 0x0004, 0x0005, 0x0006}
- Number of output clusters = 0x00

Then add attributes to this port. Since the currently created port 0x01 is active, the added attributes are added to this port. If you need to create other ports, you need to add the properties of this port before saving. If the module is restarted at this time, the active port 0x01 will be cleared. At this time, port 0x01 needs to be re-created, or other ports must be created.

**For ZCL switch equipment, the following attributes need to be added:**

| name | Cluster ID | Manufacturer code | Property ID | type of data | Support operations | Property direction |
|---|---|---|---|---|---|---|
| ZCL version | 0x0000 | 0x0000 | 0x0000 | 0x20 (uint8) | 0x01 (read only) | Sever |
| version of the agreement | 0x0000 | 0x0000 | 0x0001 | 0x20 (uint8) | 0x01 (read only) | Sever |
| Software version | 0x0000 | 0x0000 | 0x0002 | 0x20 (uint8) | 0x01 (read only) | Sever |
| hardware version | 0x0000 | 0x0000 | 0x0003 | 0x20 (uint8) | 0x01 (read only) | Sever |
| Trade Names | 0x0000 | 0x0000 | 0x0004 | 0x42 (string) | 0x01 (read only) | Sever |
| Product number | 0x0000 | 0x0000 | 0x0005 | 0x42 (string) | 0x01 (read only) | Sever |
| Production Date | 0x0000 | 0x0000 | 0x0006 | 0x42 (string) | 0x01 (read only) | Sever |
| Power supply | 0x0000 | 0x0000 | 0x0007 | 0x30 (enmu8) | 0x01 (read only) | Sever |
| property terminator | 0x0000 | 0x0000 | 0xFFFD | 0x21 (uint16) | 0x01 (read only) | Sever |
| Identity tag status | 0x0003 | 0x0000 | 0x0000 | 0x21 (uint16) | 0x03 (readable and writable) | Sever |
| property terminator | 0x0003 | 0x0000 | 0xFFFD | 0x21 (uint16) | 0x01 (read only) | Sever |
| Support grouped string naming | 0x0004 | 0x0000 | 0x0000 | 0x18 (bitmap8) | 0x01 (read only) | Sever |
| property terminator | 0x0004 | 0x0000 | 0xFFFD | 0x21 (uint16) | 0x01 (read only) | Sever |
| Cumulative number of scenes | 0x0005 | 0x0000 | 0x0000 | 0x20 (uint8) | 0x01 (read only) | Sever |
| Current scene ID | 0x0005 | 0x0000 | 0x0001 | 0x20 (uint8) | 0x01 (read only) | Sever |
| Current scene group ID | 0x0005 | 0x0000 | 0x0002 | 0x21 (uint16) | 0x01 (read only) | Sever |
| Current scene valid status | 0x0005 | 0x0000 | 0x0003 | 0x10 (bool) | 0x01 (read only) | Sever |
| Support scene string naming | 0x0005 | 0x0000 | 0x0004 | 0x18 (bitmap8) | 0x01 (read only) | Sever |
| property terminator | 0x0005 | 0x0000 | 0xFFFD | 0x21 (uint16) | 0x01 (read only) | Sever |
| switch status | 0x0006 | 0x0000 | 0x0000 | 0x10 (bool) | 0x05 (read only/report) | Sever |
| property terminator | 0x0006 | 0x0000 | 0xFFFD | 0x21 (uint16) | 0x01 (read only) | Sever |

According to the order of the table, use the "Add Attribute" command to add the attributes in the table one by one.

**Add the first attribute:**

55 0C 00 41 [00 00] [00 00] [00 00] [20] [01] [00] 60

The contents in "[ ]" are as follows

- Cluster ID = 0x0000
- Manufacturer code = 0x0000
- Property ID = 0x0000
- Data type = 0x20 (Data type 0x20 = uint8)
- Operation mode = 0x01, supports read-only
- Attribute direction = 0x00, this attribute is the server direction (controlled end direction)

Return command:

```
55 0E 00 41 [00] [01] [00 00] [00 00] [01] {[00 00] [20] [01]} 60
```

The contents in "[ ]" are as follows

- Added successfully
- Current port=0x01
- Cluster ID = 0x0000
- Manufacturer code = 0x0000
- add attribute count=1
- "{}" is the structure with added attributes, 4-byte alignment
- Property ID = 0x0000
- data type = 0x20
- Operation mode = 0x01, supports read-only

In the add attribute command, the cluster ID, manufacturer code, and attribute ID are all entered in little-endian mode. For example, to add the attribute "hardware version", you should enter the command:

```
55 0C 00 41 [00 00] [00 00] [04 00] [42] [01] [00] 06
```

The contents in "[ ]" are as follows

- Cluster ID = 0x0000
- Manufacturer code = 0x0000
- Property ID = 0x0004
- data type = 0x42 (data type 42 = string)
- Operation mode = 0x01, supports read-only
- Attribute direction = 0x00, this attribute is the server direction (controlled end direction)

Since 5 attributes have been added to cluster 0x0000 at this time, the return command is as follows:

```
55 1E 00 41 [00] [01] [00 00] [00 00] [05] {[00 00] [20] [01], [01 00] [20] [01], [02 00] [20 ] [01], [03 00] [20] [01], [04 00] [42] [01]}
02
```

The contents in "[ ]" are as follows

- Added successfully
- Current port=0x01
- Cluster ID = 0x0000
- Manufacturer code = 0x0000
- add attribute count=5
- "{}" is the structure with added attributes, 4-byte alignment
- The first attribute "[00 00] [20] [01]", attribute ID=0x0000, data type=0x20, operation mode=0x01
- The second attribute "[01 00] [20] [01]", attribute ID=0x0001, data type=0x20, operation mode=0x01
- The third attribute "[02 00] [20] [01]", attribute ID=0x0002, data type=0x20, operation mode=0x01
- The fourth attribute "[03 00] [20] [01]", attribute ID=0x0003, data type=0x20, operation mode=0x01
- The fifth attribute "[04 00] [42] [01]", attribute ID=0x0004, data type=0x42, operation mode=0x01

After adding all the attributes according to the table, save the attributes.

input the command:

```
55 03 00 42 42
```

Return command:

55 04 00 42 [00] 42

The contents in "[ ]" are as follows

- Saved successfully

After successfully saving a port and its attributes, you can continue to create a new port and add the same attributes. Since this routine needs to simulate a 3-way switch, you need to add two more identical ports. In order to ensure the compatibility of the gateway, It is recommended to use 0x02 and 0x03 as the next two ports.

## 6.1.2 Create other switch ports and add properties

**Enter the "Create ZCL Port" command to create port 2:**

55 15 00 40 [15] [02] [04 01] [00 01] [05] { [00 00] [03 00] [04 00] [05 00] [06 00] } [00] 52

The contents in "[ ]" are as follows

- The switch has a total of 0x15=21 attributes
- Create port number = 0x02
- Contour ID = 0x0104
- Device ID = 0x0100
- Input number of clusters = 0x05
- Input cluster {0x0000, 0x0003, 0x0004, 0x0005, 0x0006}
- Number of output clusters = 0x00

Return command:

55 16 00 40 [00] [01] [02] [04 01] [00 01] [05] {[00 00] [03 00] [04 00] [05 00] [06 00]} [00] 44

The contents in "[ ]" are as follows

- execution succeed
- Module assigned port index = 1
- Create port number = 0x02
- Contour ID = 0x0104
- Device ID = 0x0100
- Input number of clusters = 0x05
- Input cluster {0x0000, 0x0003, 0x0004, 0x0005, 0x0006}
- Number of output clusters = 0x00

Then continue to add all the attributes in the table, and save the attributes after adding them. Then continue to add the port number 0x03, add attributes for the port and save it.

After all attributes are added, use the reset command to reset the module, and the module will guide the newly created attributes to start.

Send reset command:

55 07 00 04 00 FF FF 00

return:

55 07 00 04 00 FF FF 00 04

Receiving an asynchronous command indicates a successful reset:

55 0D 80 00 00 [10] [E5 C7 F9 3E 7D 38 C1 A4] 55

The contents in "[]" are as follows

Module software version=0x10

Module MAC address = E5 C7 F9 3E 7D 38 C1 A4

## 6.1.3 Initialize attribute values

Initializing attribute values allows the module to be correctly identified after accessing the gateway. Each attribute created in the E180-Z5812 module can support reading and writing. Among them, E180-Z5812 sets a power-off saving value for the 8 basic attributes under cluster ID = 0x0000, and any attribute under the remaining 8 arbitrary clusters. Note that if you set the attribute value to be saved after power-off, if multiple ports support this attribute, then these attributes under all ports will become the value to be saved after power-off, which is the default value of this attribute.

**Set the default value of the attribute "Vendor Name":**

```
55 1C 00 43 [02] [00] [00] [00 00] [00 00] [04 00] [45 42 59 54 45 20 5A 69 67 42 65 65 20 33 2E 30] 31
```

The contents in "[ ]" are as follows

- Operation mode = 0x02, write attributes and save FLASH
- Port index = 0x00, the property being written is located on the first port created
- Attribute direction = 0x00, the attribute being written is the server side
- Cluster ID=0x0000, the attribute being written is located in cluster 0x0000
- Manufacturer code = 0x0000, the attribute manufacturer code being written is 0x0000
- Attribute ID=0x0004, the ID of the attribute being written is 0x0004
- The written attribute data is 16 bytes, which is the hexadecimal format of the string "EBYTE ZigBee 3.0"

Return command:

```
55 04 00 43 00 43
```

- Write successfully

Continue to set the attributes "Product Model" and "Production Date". Most gateways will verify whether the access device has this field. You can usually enter any string.

**Set the attribute "Product Model", the product model is "EBYTEOnOffSwitch"**

```
55 1C 00 43 02 00 00 00 00 00 00 05 00 45 42 59 54 45 4F 6E 4F 66 66 53 77 69 74 63 68 57
```

**Set the attribute "production date", the production date is "20231204"**

```
55 14 00 43 02 00 00 00 00 00 00 06 00 32 30 32 33 31 32 30 34 43
```

Then reset the module so that the same attribute values are applied to all three ports.

## 6.1.4 Access gateway

Use the APP provided by the gateway to add devices. Usually the APP has the function of scanning to add any device and adding a specified device. Both methods can be added.

After the APP turns on the device addition mode, send the "Start Network Configuration" command to the module. Before configuring the network, you must first confirm that the module is a router, terminal node or dormant terminal. The terminal node and dormant terminal can be switched through "Set Node Type" .

Send "Start Network Configuration Command"

```
55 03 00 02 02
```

Return command:

```
55 04 00 02 00 02
```

The module begins to enter the network configuration mode and waits for 4 seconds before returning the asynchronous command.

```
55 29 80 01 [02] [E5 C7 F9 3E 7D 38 C1 A4] [0B] [26 B1] [4C 91] [55 A1 42 ED B3 6D B3 66] [E5 0F F3 F9 9A CC 9E 13 28 3C DF 2E DE 65 08 2E] 14
```

The contents in "[ ]" are as follows

- Network distribution successful
- MAC address = E5 C7 F9 3E 7D 38 C1 A4
- Gateway channel=0x0B, channel 11, frequency 2405MHz
- Gateway PANID=0xB126
- Short address=0x914C
- Extended PANID=55 A1 42 ED B3 6D B3 66
- Gateway's NWK layer secret key = E5 0F F3 F9 9A CC 9E 13 28 3C DF 2E DE 65 08 2E

At the same time, on the APP interface, an interface for searching for new devices appears, which is a three-way switch.


## 6.1.5 Use APP to control simulated 3-way switch

Enter the 3-way switch interface of the APP and trigger switch 2.


The module receives the ZCL control command:

55 0F 82 0F [A1] [00 00] [01] [66] [00] [06 00] [00 00] [DE] [01] 92

The contents in "[ ]" are as follows

- Receive mode = 0xA1, the received signal strength is valid, and there is no need to reply to the control command of the default return command. The local port index 1 receives the command. According to the record when the port was created, port index 1 corresponds to port 0x02.
- Source short address = 0x0000, this control command is sent by the coordinator.
- Source port = 0x01, this command is sent by the coordinator using port 0x01.
- Frame number=0x66
- Command direction = 0x00, which is a Client->Sever command, that is, the command sent by the control end to the controlled end.
- Cluster ID=0x0006, which is a switch control command.
- Manufacturer code=0x0000
- Signal strength RSSI=0xDE, converted to -33dbm
- Command ID=0x01, the command ID=0x01 located under cluster ID=0x0006, the corresponding command is "turn on the switch", this command has no command parameters.

After receiving this command, change the attribute "switch status" under port index 1 to "1". This attribute is located at attribute ID=0x0000 of cluster ID=0x0006, so use the command "read-write attribute" to modify the value of this attribute. Note that the switch status must not be saved to FLASH as the default value.

**Send read and write attribute commands:**

55 0D 00 43 [01] [01] [00] [06 00] [00 00] [00 00] [01] 44

- Operation mode = 0x01, write attribute operation does not save FLASH
- Port index = 0x01, the property being written is on the second port created (index 1)
- Attribute direction = 0x00, the attribute being written is the server side

- Cluster ID=0x0006, the attribute being written is located in cluster 0x0006
- Manufacturer code = 0x0000, the attribute manufacturer code being written is 0x0000
- Attribute ID=0x0000, the ID of the attribute being written is 0x0000
- Attribute value=0x01

Since the "switch status" attribute is set to be automatically reported when added, after the attribute value is modified, it will be synchronized to the APP.


# 6.2 Example 2: Simulating human body infrared sensor access gateway

## 6.2.1 Create sensor port and add related properties

Create a 1-channel human body infrared sensor, and use 0x01 for the port number.

First, exit the module from the network. You can use the "Reset/Factory Restore" command to exit the network, or you can initiate a device deletion operation from the APP.

Send the "Clear Ports and Attributes" command

```
55 03 00 44 44
```

return command

```
55 04 00 44 00 44
```

This command indicates successful execution.

**Create a sensor port:**

```
55 11 00 40 [11] [01] [04 01] [02 04] [03] {[00 00] [03 00] [00 05]} [00] 56
```

The contents in "[ ]" are as follows

- The switch has a total of 0x11=17 attributes.

- Create port number = 0x01
- Contour ID = 0x0104
- Device ID = 0x0402, corresponding to security sensor equipment
- Input number of clusters = 0x03
- Input cluster {0x0000, 0x0003, 0x0500}
- Number of output clusters = 0x00

Return command:

55 12 00 40 [00] [00] [01] [04 01] [02 04] 03 00 00 03 00 00 05 00 47

The contents in "[ ]" are as follows

- execution succeed
- Module assigned port index = 0
- Create port number = 0x01
- Contour ID = 0x0104
- Device ID = 0x0402
- Input number of clusters = 0x03
- Input cluster {0x0000, 0x0003, 0x0500}
- Number of output clusters = 0x00

For ZCL security sensor equipment, the following attributes need to be added:

| name | Cluster ID | Manufacturer code | Property ID | type of data | Support operations | Property direction |
|---|---|---|---|---|---|---|
| ZCL version | 0x0000 | 0x0000 | 0x0000 | 0x20 (uint8) | 0x01 (read only) | Sever |
| version of the agreement | 0x0000 | 0x0000 | 0x0001 | 0x20 (uint8) | 0x01 (read only) | Sever |
| Software version | 0x0000 | 0x0000 | 0x0002 | 0x20 (uint8) | 0x01 (read only) | Sever |
| hardware version | 0x0000 | 0x0000 | 0x0003 | 0x20 (uint8) | 0x01 (read only) | Sever |
| Trade Names | 0x0000 | 0x0000 | 0x0004 | 0x42 (string) | 0x01 (read only) | Sever |
| Product number | 0x0000 | 0x0000 | 0x0005 | 0x42 (string) | 0x01 (read only) | Sever |
| Production Date | 0x0000 | 0x0000 | 0x0006 | 0x42 (string) | 0x01 (read only) | Sever |
| Power supply | 0x0000 | 0x0000 | 0x0007 | 0x30 (enmu8) | 0x01 (read only) | Sever |
| property terminator | 0x0000 | 0x0000 | 0xFFFD | 0x21 (uint16) | 0x01 (read only) | Sever |
| Identity tag status | 0x0003 | 0x0000 | 0x0000 | 0x21 (uint16) | 0x03 (readable and writable) | Sever |
| property terminator | 0x0003 | 0x0000 | 0xFFFD | 0x21 (uint16) | 0x01 (read only) | Sever |
| Security registration status | 0x0500 | 0x0000 | 0x0000 | 0x30 (enmu8) | 0x01 (read only) | Sever |
| Security sensor type | 0x0500 | 0x0000 | 0x0001 | 0x31 (enmu16) | 0x01 (read only) | Sever |
| Security alarm status | 0x0500 | 0x0000 | 0x0002 | 0x19 (bitmap16) | 0x01 (read only) | Sever |
| Alarm reporting MAC address | 0x0500 | 0x0000 | 0x0010 | 0xF0 (EUI 64) | 0x03 (readable and writable) | Sever |
| Zone ID | 0x0500 | 0x0000 | 0x0011 | 0x20 (uint8) | 0x01 (read only) | Sever |

## 6.2.2 Set attribute initialization values and access the gateway

After adding the above attributes, save and reset and restart, then modify the attributes "Manufacturer Name", "Product Model", and "Production Date". Just enter a valid string.

For security sensors, you need to edit the attribute "Security Sensor Type". According to the ZCL specification table, the corresponding value of "Motion sensor" is 0x000D. Modify the attribute value and save it in FLASH.

Enter the read and write attribute command to modify the "Security Sensor Type"

```
55 0E 00 43 [02] [00] [00] [00 05] [00 00] [01 00] [0D 00] 48
```

The contents in "[ ]" are as follows

- Operation mode = 0x02, write attributes and save FLASH
- Port index = 0x00, the property being written is located on the first port created
- Attribute direction = 0x00, the attribute being written is the server side
- Cluster ID=0x0500, the attribute being written is located in cluster 0x0500
- Manufacturer code = 0x0000, the attribute manufacturer code being written is 0x0000
- Attribute ID=0x0001, the ID of the attribute being written is 0x0001
- Written attribute value = 0x000D

Then reset the module and add the module to the gateway.


## 6.2.3 Sensor triggers alarm

**Send ZCL control command:**

```
55 15 02 0F [00] [00 00] [01] [75] [01] [00 05] [00 00] [00] [00] [01 00 00 FE 01 00] 83
```

The contents in "[ ]" are as follows

- Send mode = 0x00, port index 0 sends command, no mode.
- Target short address=0x0000
- Destination port=0x01
- Frame number=0x75
- Command direction=Server->Client, this is a command sent from the controlled end to the control end, and the sensors are all controlled ends.
- Cluster ID=0x0500

- Manufacturer code=0x0000
- Reply mode=0
- Command ID=0x00
- Command parameter=[01 00 00 FE 01 00]

Analyze the command parameters according to "Zigbee Cluster Library". The parameters of the security alarm command are as shown in the table.

| Bits | 16 | 8 | 8 | 16 |
|---|---|---|---|---|
| Data Type | map16 | map8 | uint8 | uint16 |
| Field Name | Zone Status | Extended Status | Zone ID | Delay |

**Command parameters:**

[01 00] [00] [FE] [01 00]

The contents in "[ ]" are as follows

- Alarm status = 0x0001. According to the sensor protocol specification, you also need to write the attribute "security alarm status" to this value.
- extended status=0x00
- Defense zone ID=0xFE, this value also needs to be written simultaneously with the attribute "defense zone ID"
- Alarm delay=0x0001, unit 250ms.

Attachment: Notes on sensor trigger control commands:

- Sensors are mostly dormant terminals and have a high failure rate in sending commands. Each time the alarm command is retransmitted, an "alarm delay" needs to be accumulated based on time.
- The "alarm status" and "defense zone ID" in the alarm command need to be synchronized with the attributes "security alarm status" and "defense zone ID" in the device. It is recommended that when the MCU detects an alarm during operation, write these two attribute values first, and then Read these two attributes and send the alarm command regularly with a cycle of 250ms. The timing will not stop until the alarm command is sent successfully.

**Receive feedback command after issuing ZCL control command:**

55 05 02 0F [00] 75 78

- Send command is valid

**Receive sending confirmation command:**

55 0A 8F 02 [00] [00 00] [01] [75] [01] [00] F8

The contents in "[ ]" are as follows

- Send mode = 0x00, port index 0 sends command, no mode.
- Target short address=0x0000
- Destination port=0x01
- Frame number=0x75
- Command direction=Server->Client, this is a command sent from the controlled end to the control end, and the sensors are all controlled ends.
- Sending result=0x00, sending successfully

**Because the default return frame was not turned off when sending, the default return frame from the coordinator was also received.**

55 10 82 0B [20] [00 00] [01] [75] [00] [00 05] [00 00] [C8] [00] [00] 10

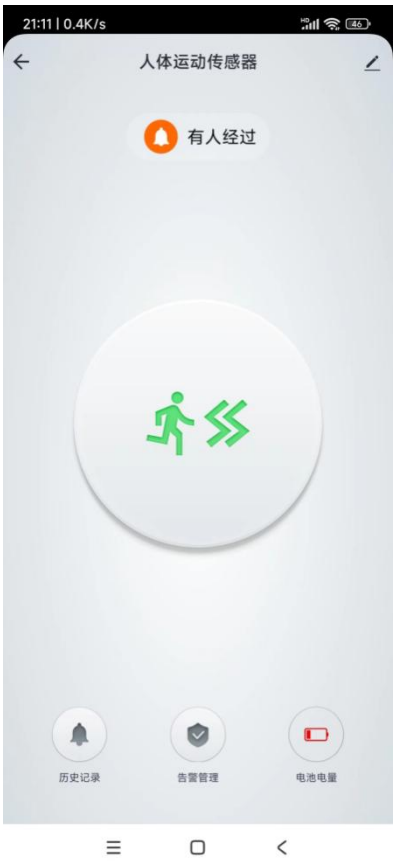
The contents in "[ ]" are as follows

- Receive mode = 0x20, the received signal strength is valid, and the local port index 0 is port 0x02 that receives the

command.

- Source short address = 0x0000, this control command is sent by the coordinator.
- Source port = 0x01, this command is sent by the coordinator using port 0x01.
- Frame number = 0x75, which is the same as the frame number of the ZCL control command sent.
- Command direction = 0x00, which is a Client->Sever command, that is, the command sent by the control end to the controlled end.
- Cluster ID=0x0500, which is a security sensor class command.
- Manufacturer code=0x0000
- Signal strength RSSI=0xC8, converted to -56dbm
- Command ID=0x00, corresponding to the control command just sent
- ZCL status = 0x00, the command just sent was correctly received and processed by the gateway.

After the command is issued, the sensor alarm is displayed on the APP.




# 7 FAQ

## 7.1 Transmission distance is not ideal

- When there are straight-line communication obstacles, the communication distance will be correspondingly attenuated;

- Temperature, humidity, and co-channel interference will cause the communication packet loss rate to increase;
- The ground absorbs and reflects radio waves, and the test effect is poor when close to the ground;
- Seawater has a strong ability to absorb radio waves, so the seaside test results are poor;
- If there are metal objects near the antenna, or if it is placed in a metal case, the signal attenuation will be very serious;
- The power register setting is wrong and the air rate is set too high (the higher the air rate, the closer the distance);
- The low voltage of the power supply at room temperature is lower than the recommended value. The lower the voltage, the smaller the power generated;
- There is a poor match between the antenna and the module or there is a problem with the quality of the antenna itself.

## 7.2 Modules are easily damaged

- Please check the power supply to ensure that it is within the recommended power supply voltage. If it exceeds the maximum value, it will cause permanent damage to the module;
- Please check the stability of the power supply. The voltage cannot fluctuate greatly and frequently;
- Please ensure anti-static operation during installation and use, as high-frequency devices are sensitive to static electricity;
- Please ensure that the humidity during installation and use should not be too high, as some components are humidity-sensitive devices;
- If there are no special needs, it is not recommended to use it at too high or too low temperature.

## 7.3 Bit error rate is too high

- If there is co-channel signal interference nearby, stay away from the interference source or modify the frequency or channel to avoid interference;
- Unsatisfactory power supply may also cause garbled code, so be sure to ensure the reliability of the power supply;
- Poor quality or too long extension cords and feeders can also cause a high bit error rate.

# Revise history

| Version | Revision date | Revision Notes | Maintenance man |
|---|---|---|---|
| 1.0 | 2024-01-04 | initial version | Bin |
| | | | |

# About us

Technical support: support@cdebyte.com

Documents and RF Setting download link: https://www.cdebyte.com

Thank you for using Ebyte products! Please contact us with any questions or suggestions: info@cdebyte.com

---

Address: B5 Mould Park, 199# Xiqu Ave, High-tech District, Sichuan, China

Chengdu Ebyte Electronic Technology Co.,Ltd.